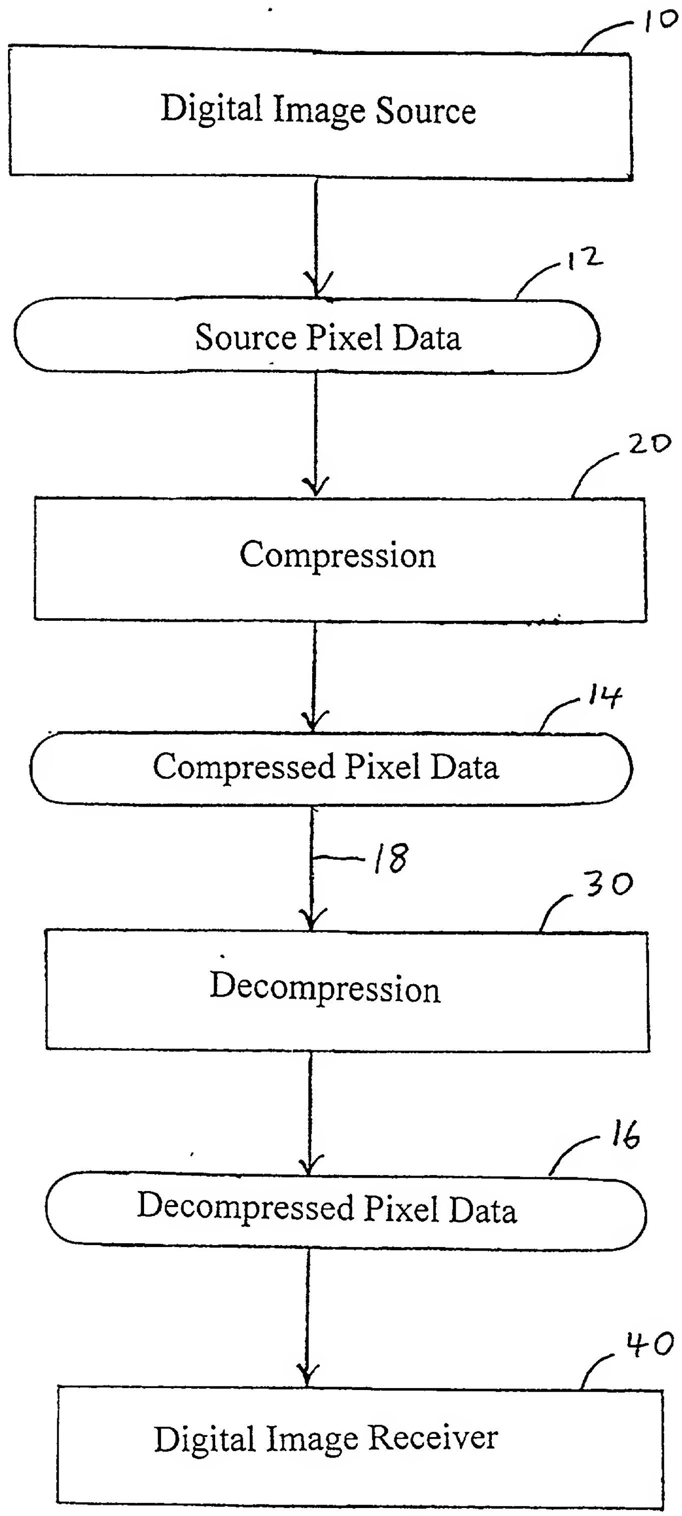
10
Digital Image Source
12
Source Pixel Data
20
Compression
14
Compressed Pixel Data
18
30
Decompression
16
Decompressed Pixel Data
40
Digital Image Receiver

FIG. 1

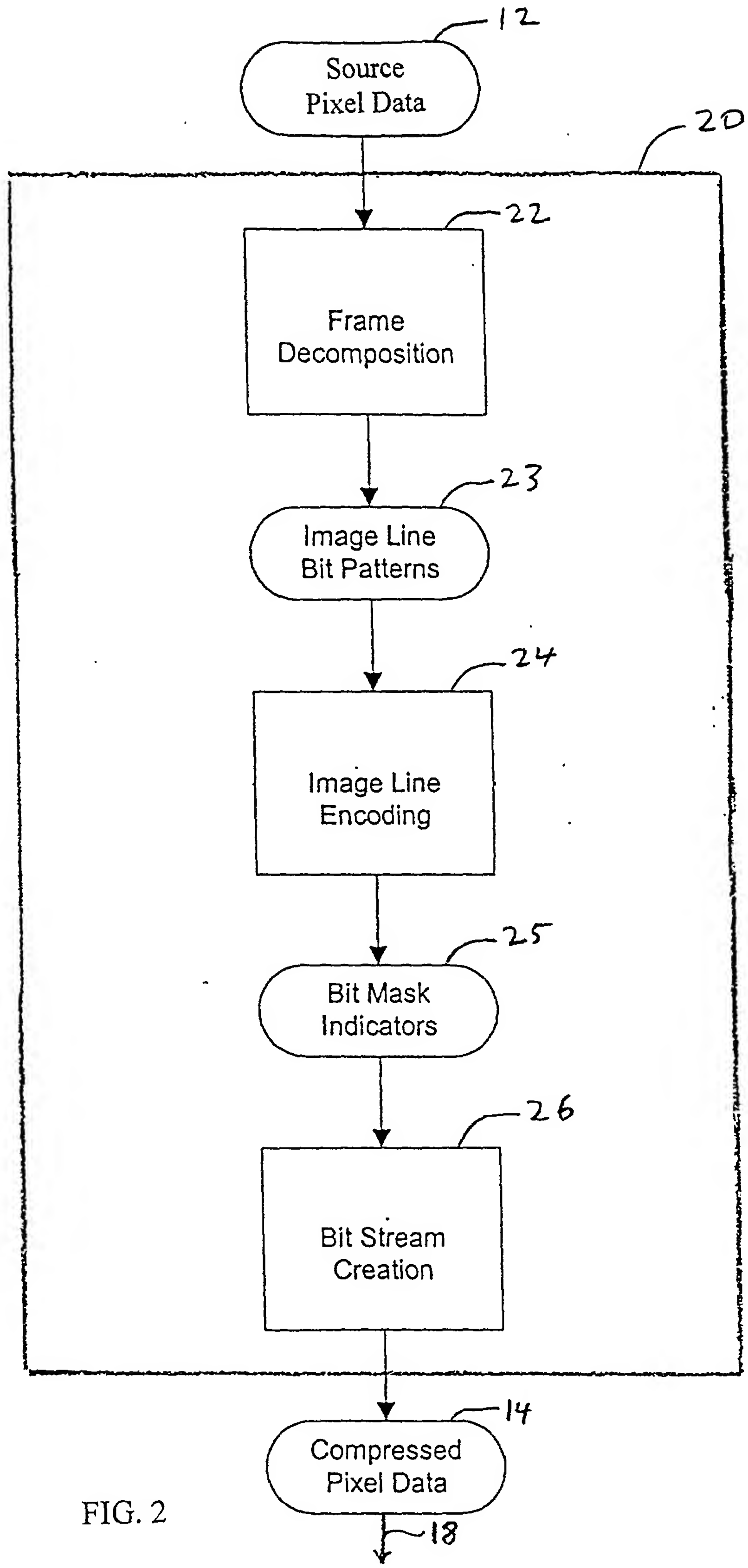
12
Source
Pixel Data
20
22
Frame
Decomposition
23
Image Line
Bit Patterns
24
Image Line
Encoding
25
Bit Mask
Indicators
26
Bit Stream
Creation
14
Compressed
Pixel Data
18

FIG. 2

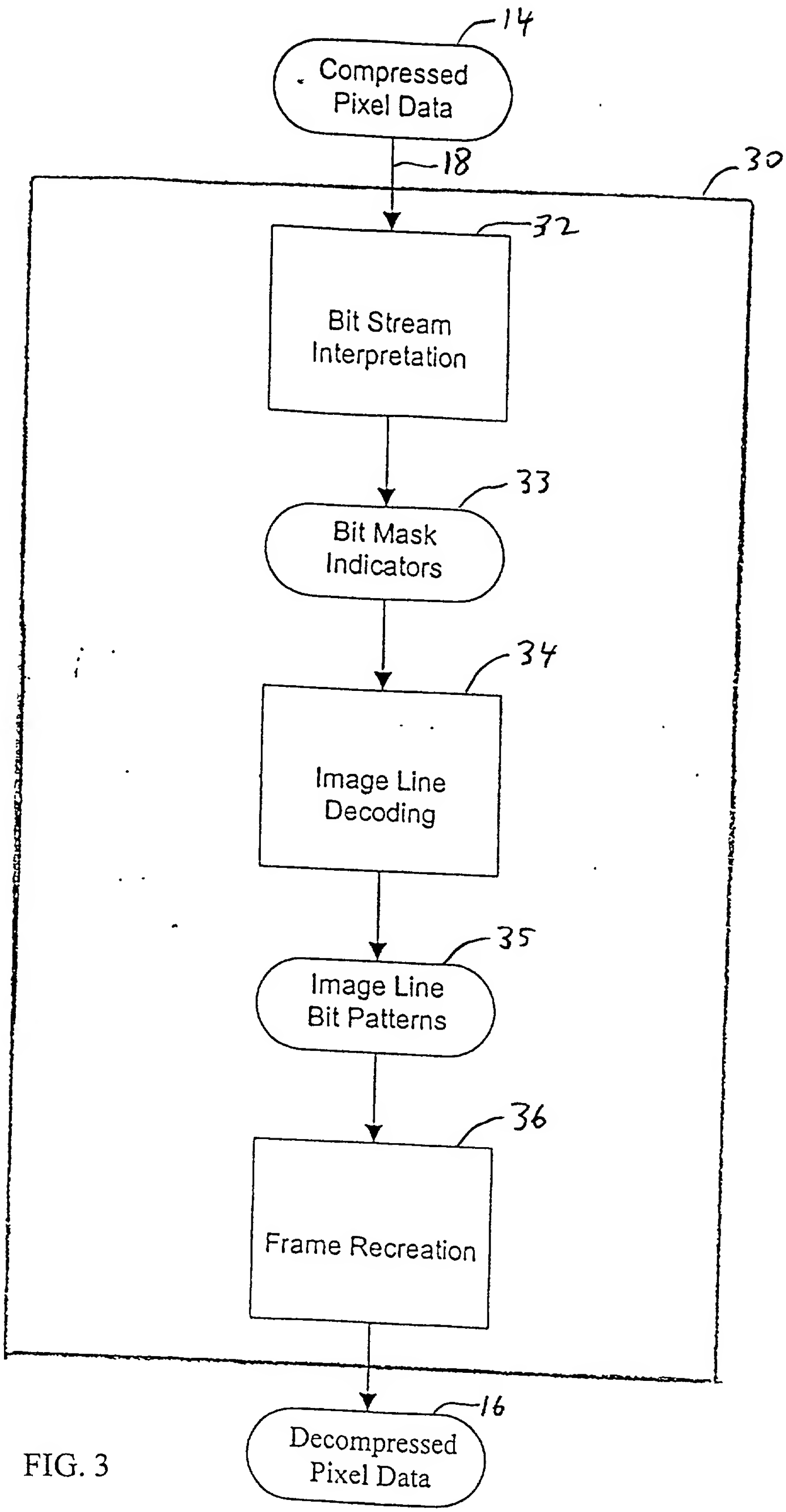
14
Compressed Pixel Data
18
30
32
Bit Stream Interpretation
33
Bit Mask Indicators
34
Image Line Decoding
35
Image Line Bit Patterns
36
Frame Recreation
16
Decompressed Pixel Data

FIG. 3

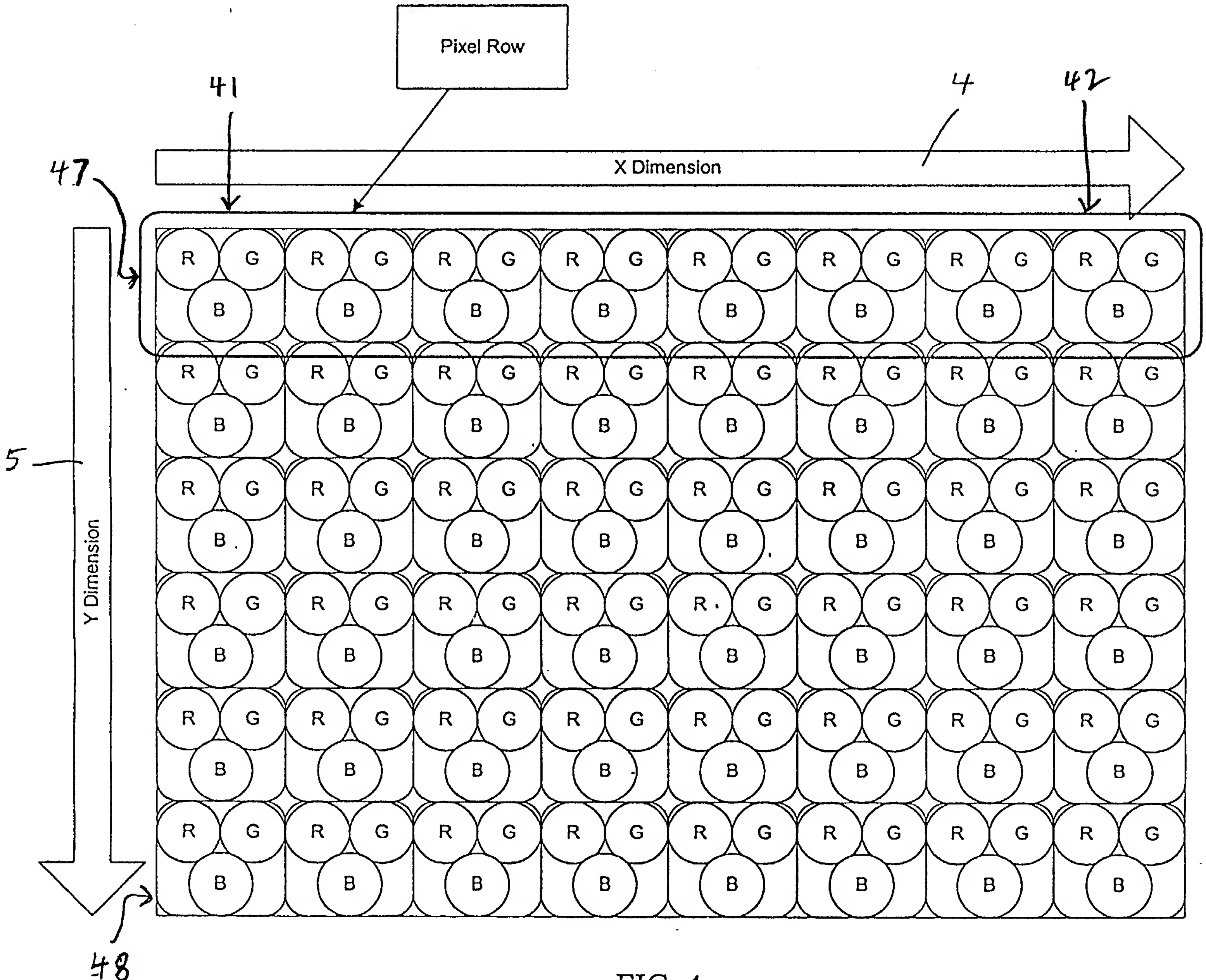

FIG. 4

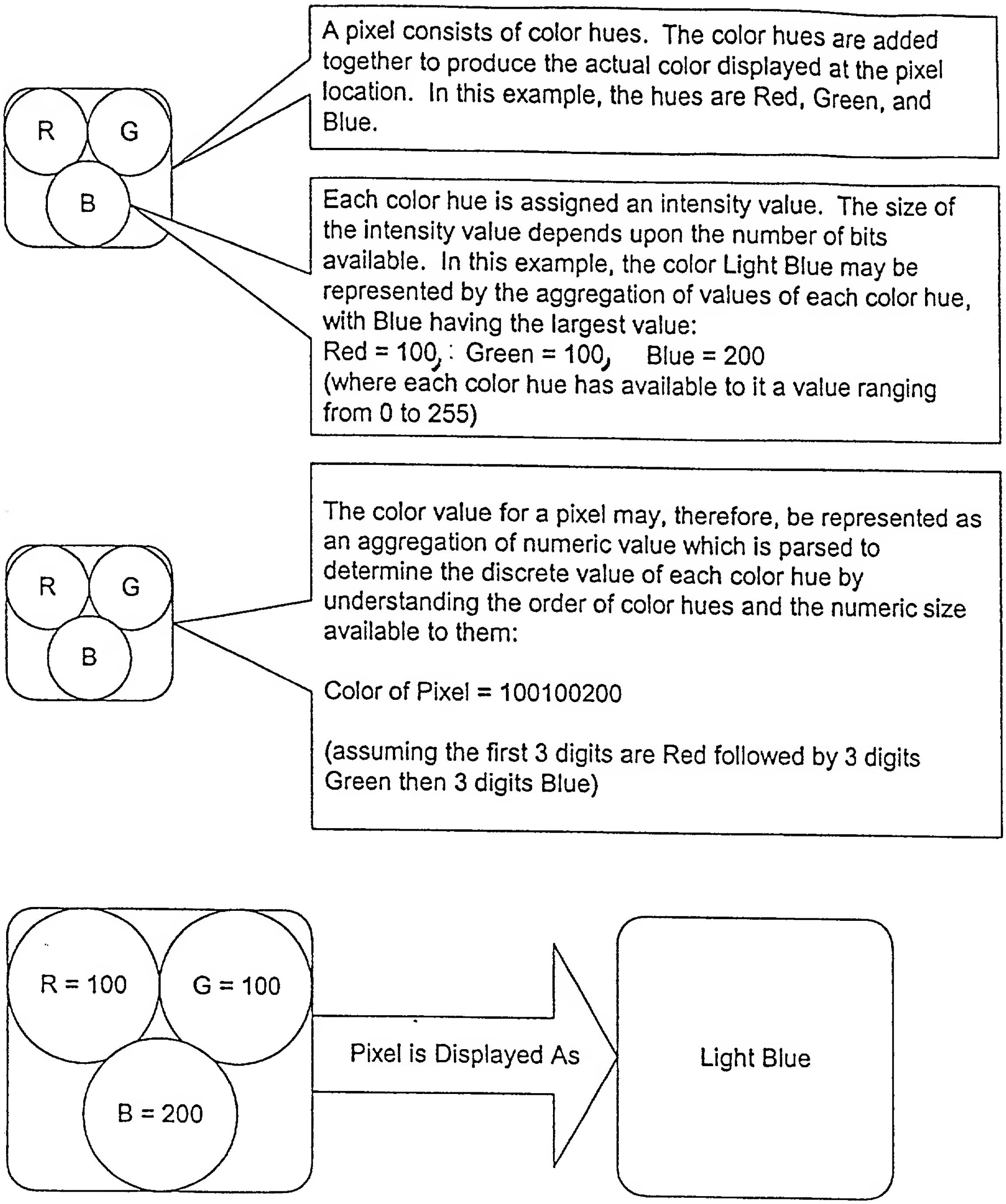
A pixel consists of color hues. The color hues are added together to produce the actual color displayed at the pixel location. In this example, the hues are Red, Green, and Blue.
R
G
B
Each color hue is assigned an intensity value. The size of the intensity value depends upon the number of bits available. In this example, the color Light Blue may be represented by the aggregation of values of each color hue, with Blue having the largest value:
Red = 100, Green = 100, Blue = 200
(where each color hue has available to it a value ranging from 0 to 255)
R
G
B
The color value for a pixel may, therefore, be represented as an aggregation of numeric value which is parsed to determine the discrete value of each color hue by understanding the order of color hues and the numeric size available to them:
Color of Pixel = 100100200
(assuming the first 3 digits are Red followed by 3 digits Green then 3 digits Blue)
R = 100
G = 100
B = 200
Pixel is Displayed As
Light Blue

FIG. 5

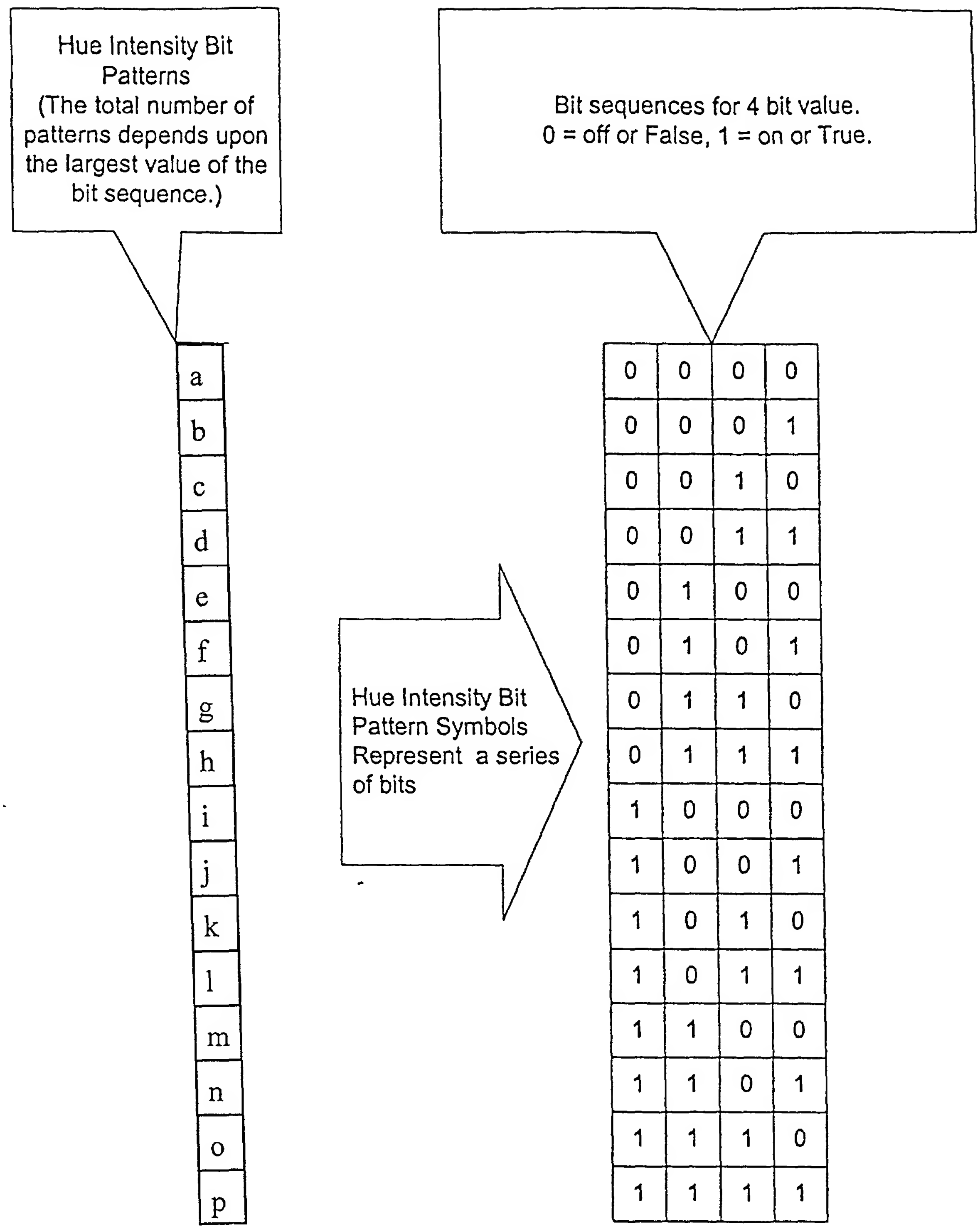
Hue Intensity Bit Patterns
(The total number of patterns depends upon the largest value of the bit sequence.)
Bit sequences for 4 bit value.
0 = off or False, 1 = on or True.
Hue Intensity Bit Pattern Symbols Represent a series of bits
a
b
c
d
e
f
g
h
i
j
k
l
m
n
o
p
0 0 0 0
0 0 0 1
0 0 1 0
0 0 1 1
0 1 0 0
0 1 0 1
0 1 1 0
0 1 1 1
1 0 0 0
1 0 0 1
1 0 1 0
1 0 1 1
1 1 0 0
1 1 0 1
1 1 1 0
1 1 1 1

FIG. 6

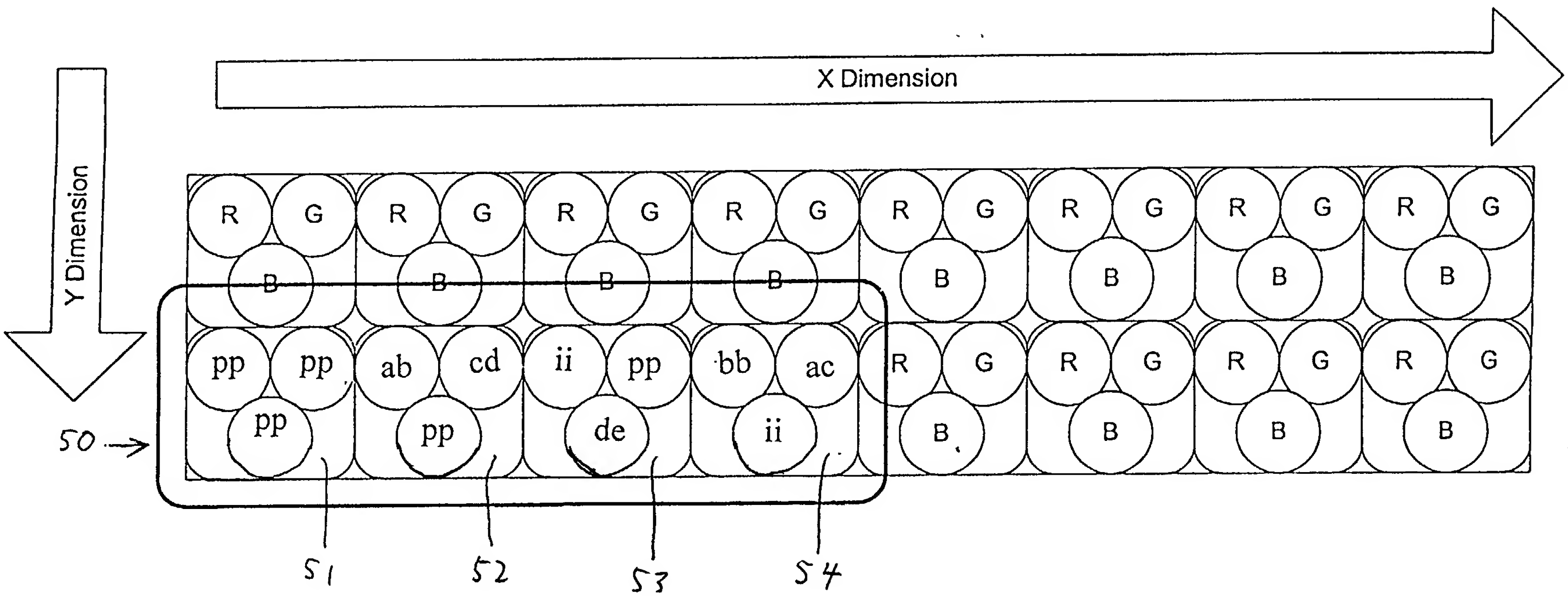
X Dimension
Y Dimension
R
G
B
pp
pp
pp
ab
cd
pp
ii
pp
de
bb
ac
ii
50
51
52
53
54

FIG. 7

52

	Bit Positions For Color Hues					
↓HIBP*	0	1	2	3	4	5
a	1					
b		1				
c			1			
d				1		
e						
f						
g						
h						
i			44			
j						
k						
l						
m						
n						
o						
p					1	1

* HIBP: Hue Intensity Bit Pattern

FIG. 8

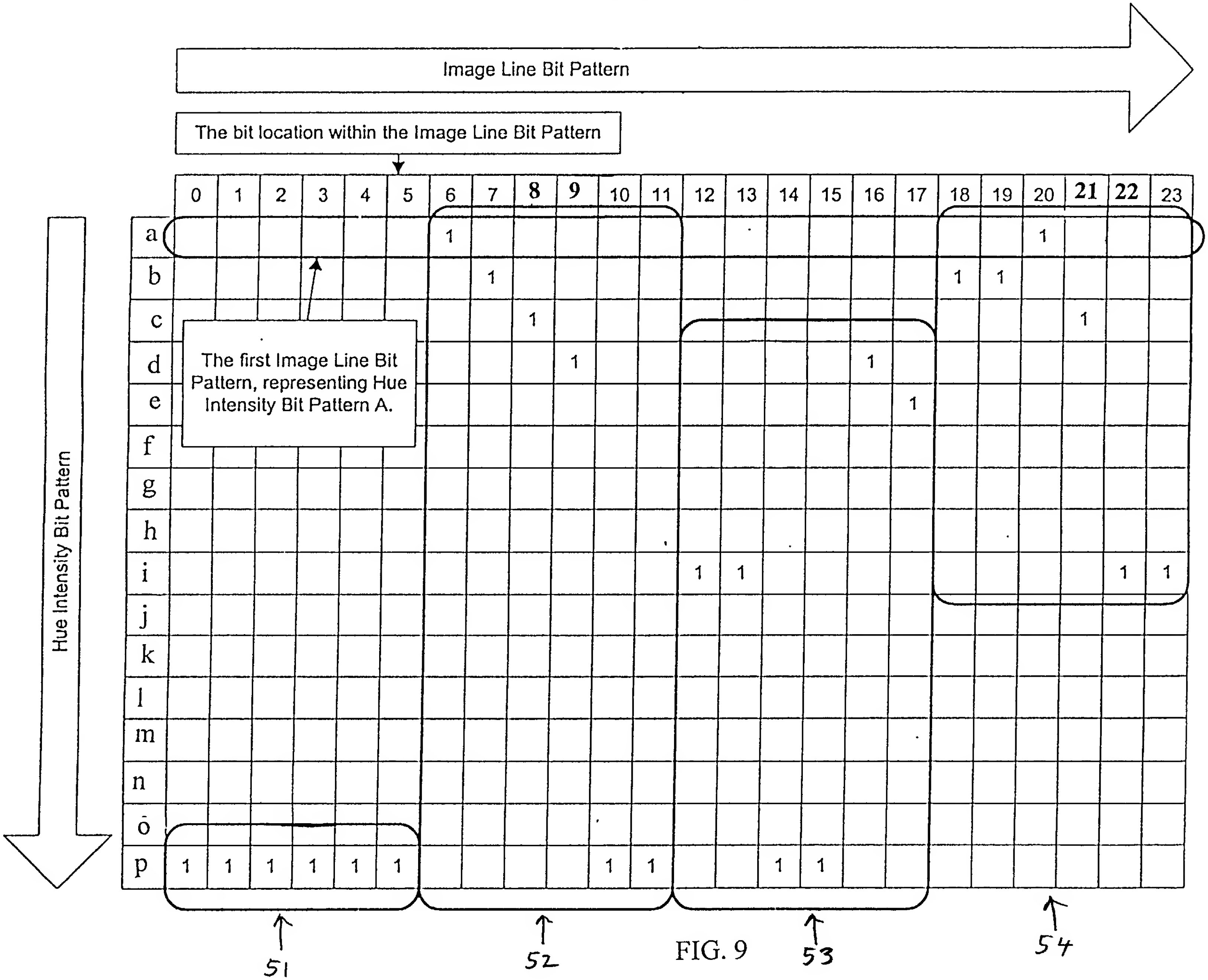
Image Line Bit Pattern
The bit location within the Image Line Bit Pattern
The first Image Line Bit Pattern, representing Hue Intensity Bit Pattern A.
Hue Intensity Bit Pattern
51
52
53
54

FIG. 9

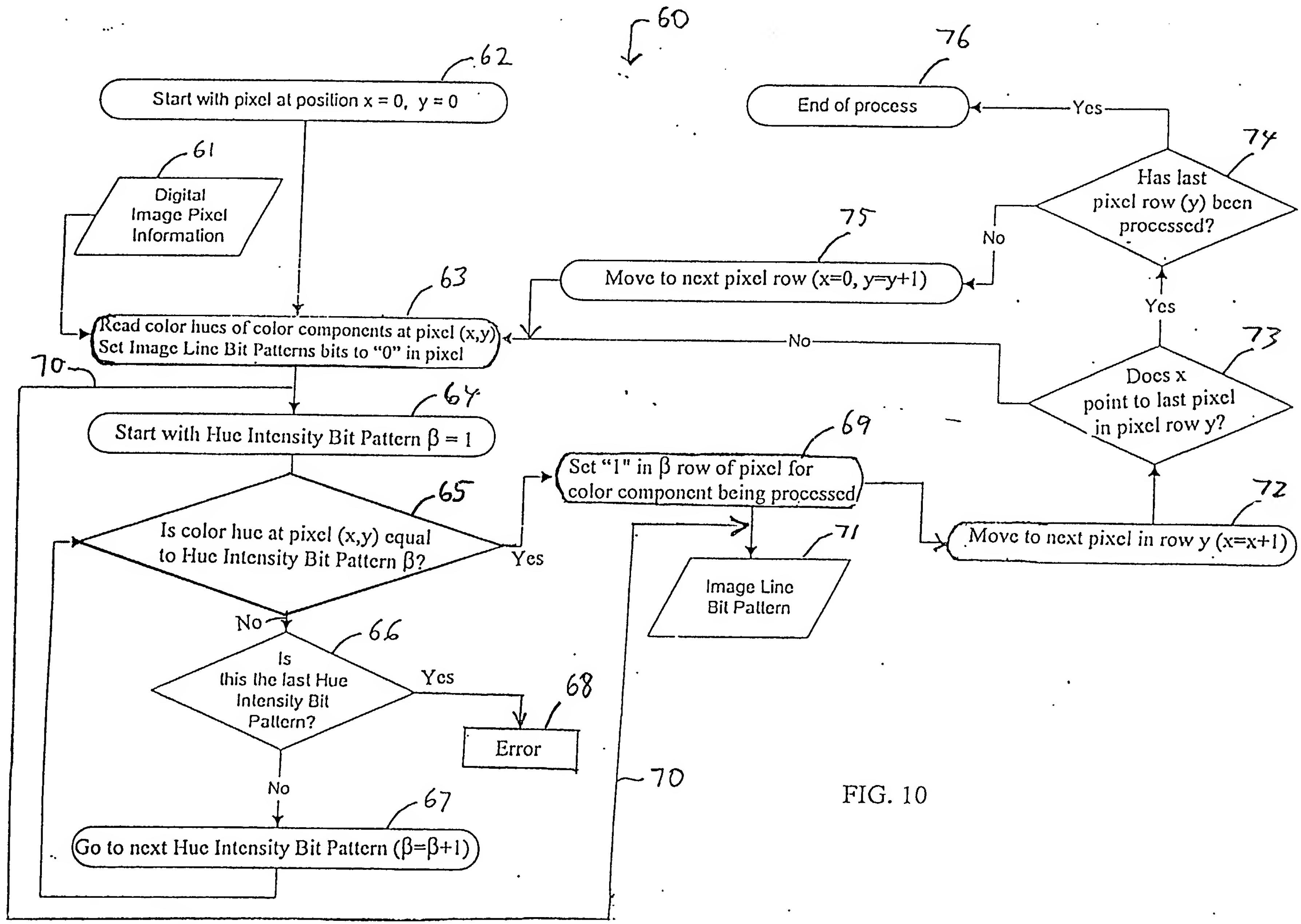
60
62
Start with pixel at position x = 0, y = 0
61
Digital Image Pixel Information
63
Read color hues of color components at pixel (x,y)
Set Image Line Bit Patterns bits to "0" in pixel
64
Start with Hue Intensity Bit Pattern β = 1
65
Is color hue at pixel (x,y) equal to Hue Intensity Bit Pattern β?
Yes
No
66
Is this the last Hue Intensity Bit Pattern?
Yes
68
Error
No
67
Go to next Hue Intensity Bit Pattern (β=β+1)
70
70
69
Set "1" in β row of pixel for color component being processed
71
Image Line Bit Pattern
72
Move to next pixel in row y (x=x+1)
73
Does x point to last pixel in pixel row y?
Yes
No
74
Has last pixel row (y) been processed?
Yes
No
75
Move to next pixel row (x=0, y=y+1)
76
End of process

FIG. 10

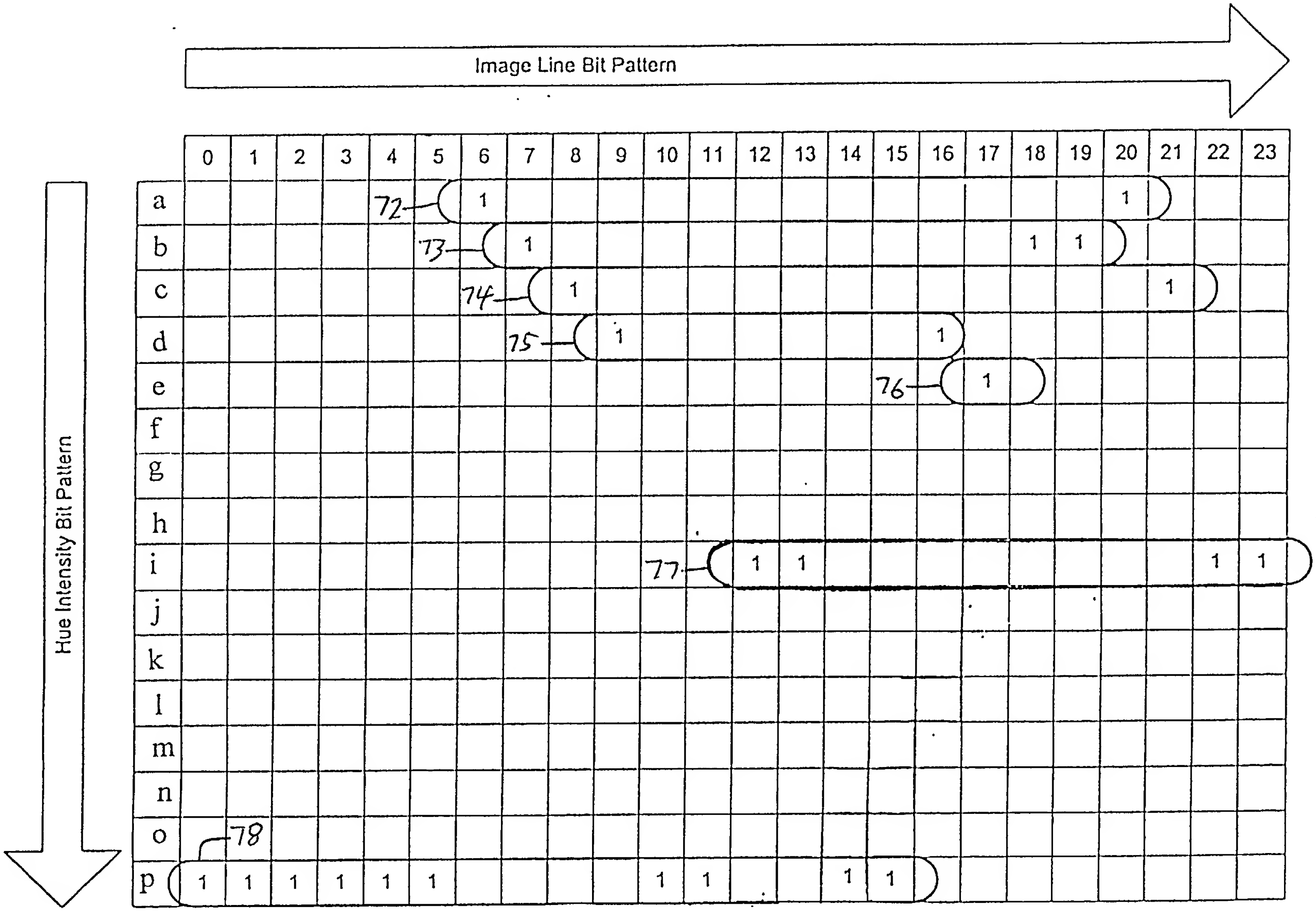

Image Line Bit Pattern
Hue Intensity Bit Pattern
72
73
74
75
76
77
78

FIG. 11

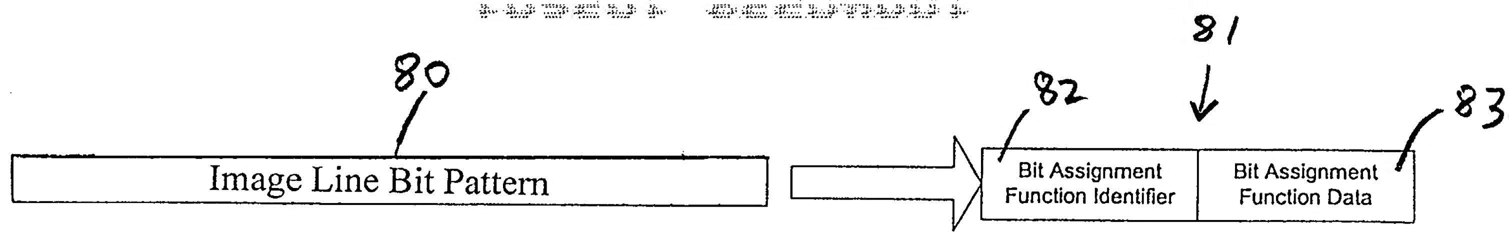

FIG. 12A

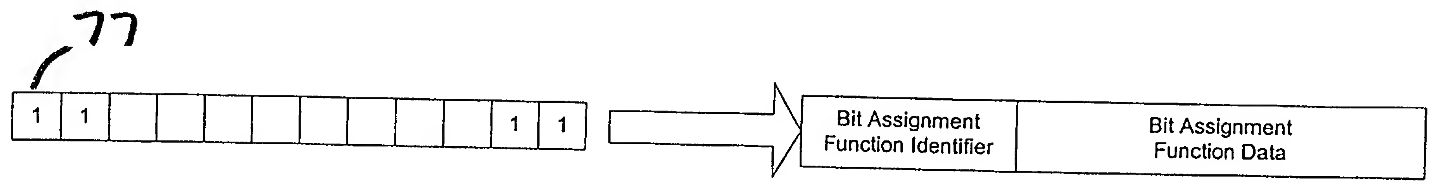

FIG. 12B

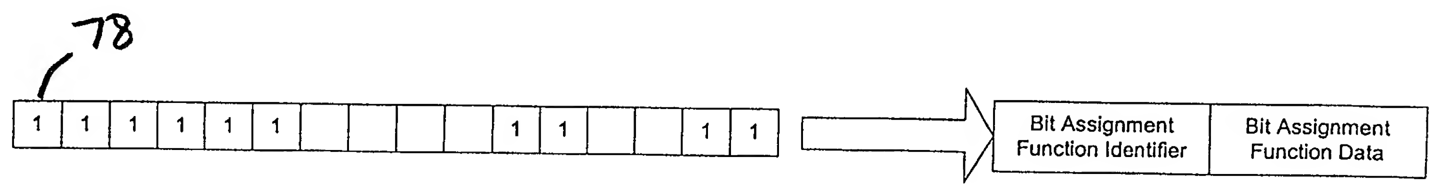

FIG. 12C

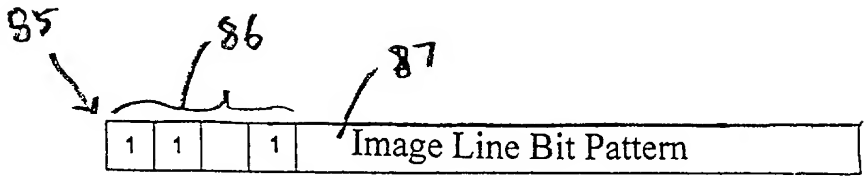

FIG. 13

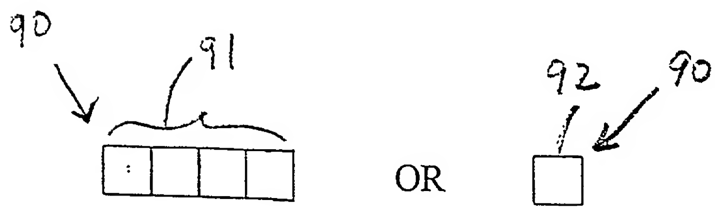

FIG. 14

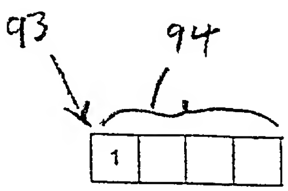

FIG. 15

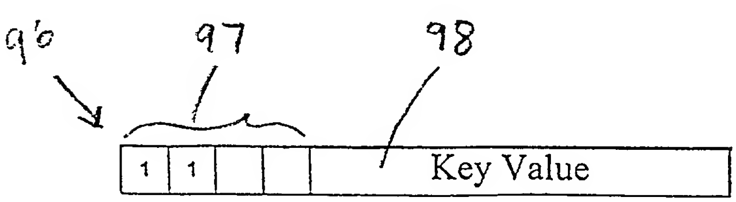

FIG. 16

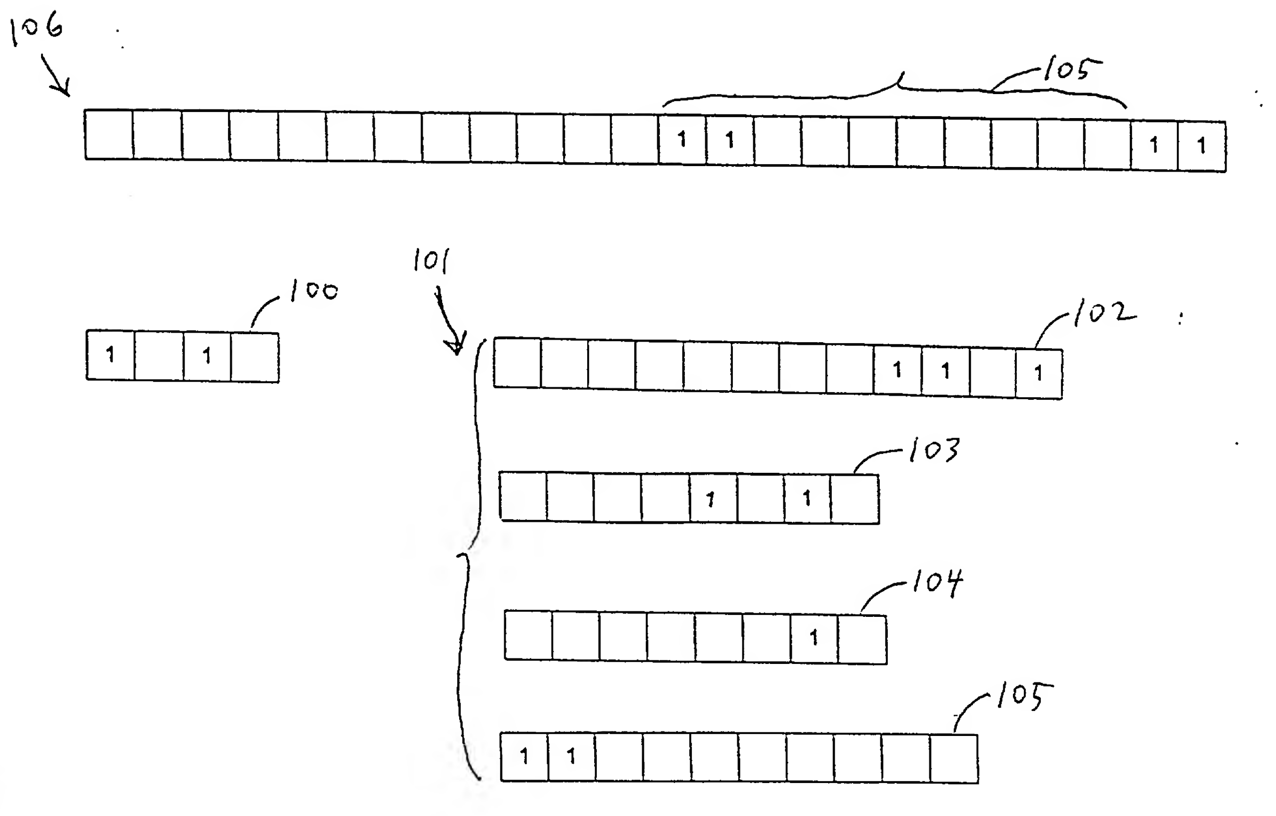

FIG. 17A

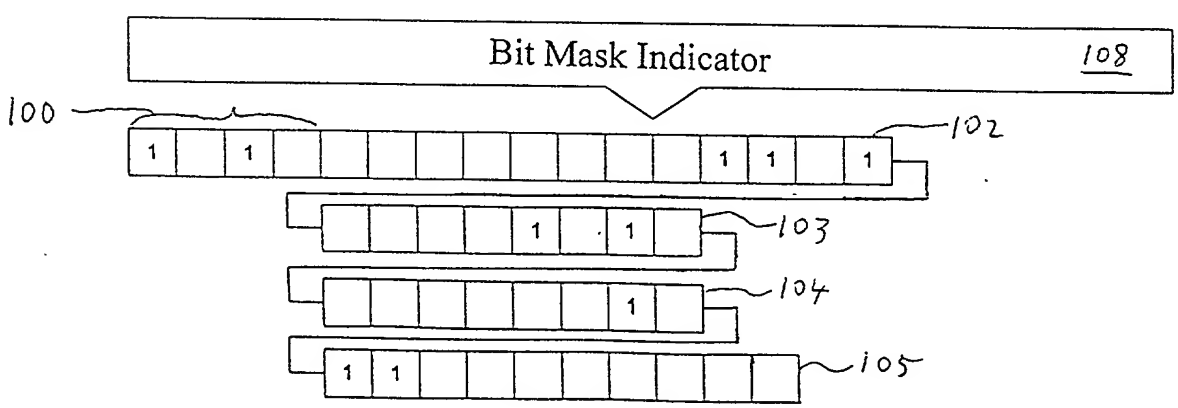

FIG. 17B

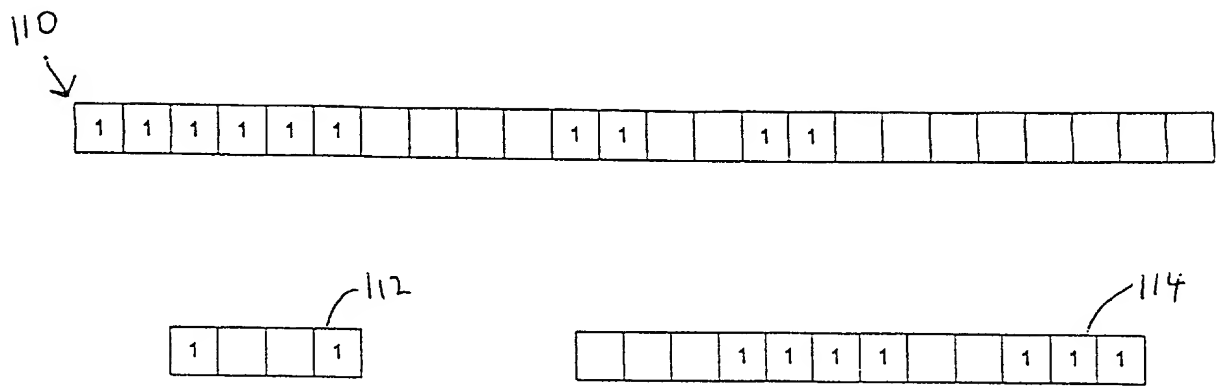

FIG. 18A

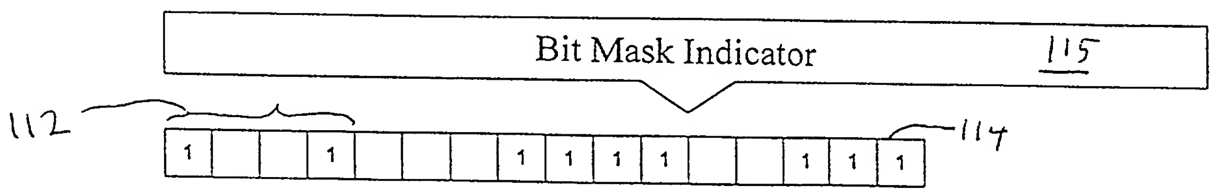

FIG. 18B

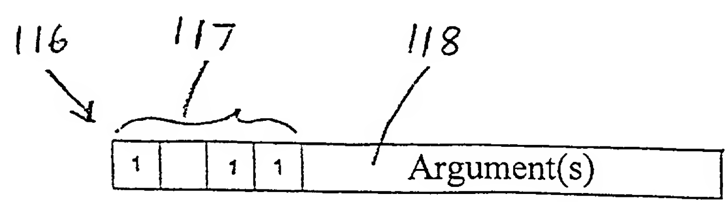

FIG. 19

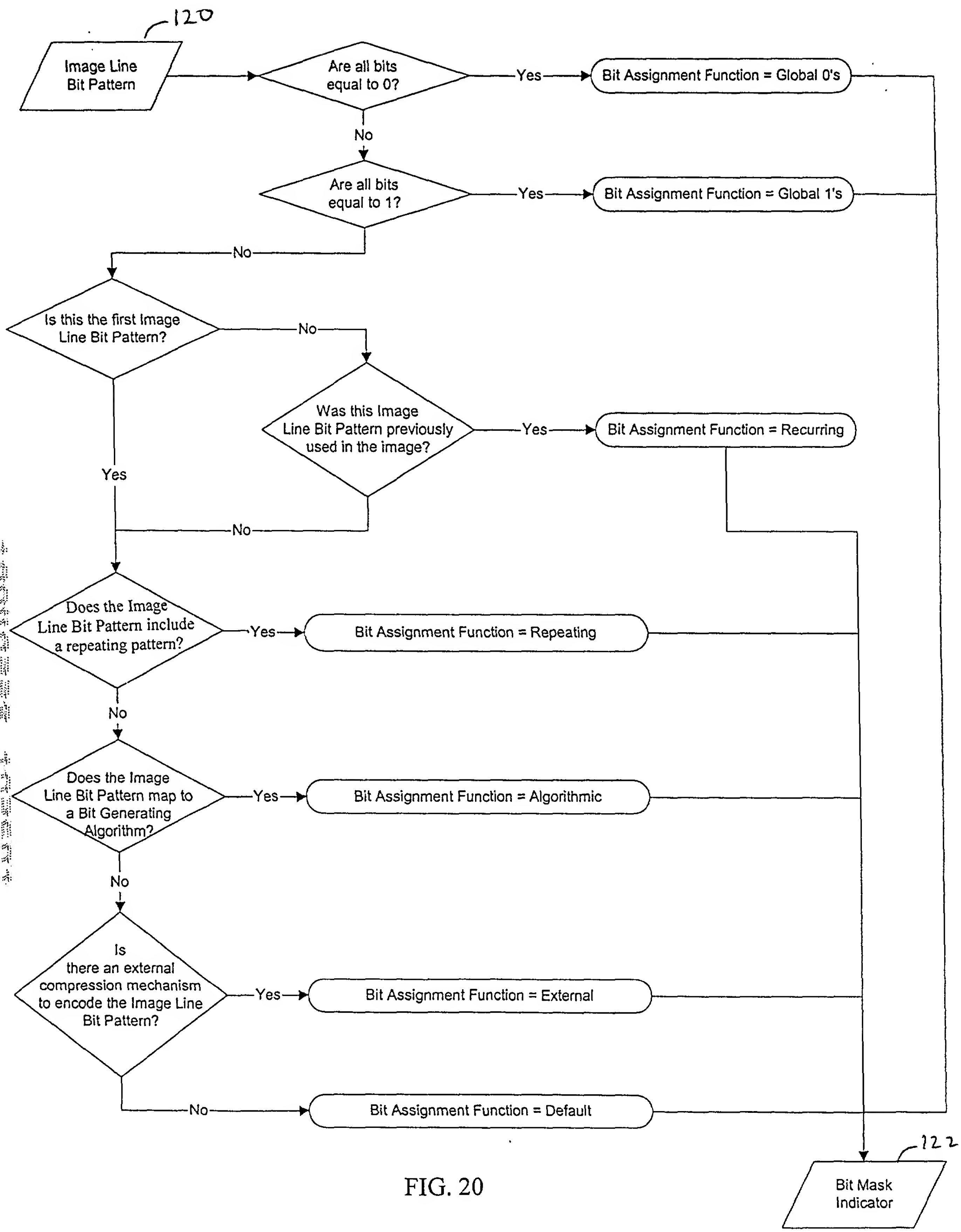
120
Image Line Bit Pattern
Are all bits equal to 0?
Yes
Bit Assignment Function = Global 0's
No
Are all bits equal to 1?
Yes
Bit Assignment Function = Global 1's
No
Is this the first Image Line Bit Pattern?
No
Was this Image Line Bit Pattern previously used in the image?
Yes
Bit Assignment Function = Recurring
Yes
No
Does the Image Line Bit Pattern include a repeating pattern?
Yes
Bit Assignment Function = Repeating
No
Does the Image Line Bit Pattern map to a Bit Generating Algorithm?
Yes
Bit Assignment Function = Algorithmic
No
Is there an external compression mechanism to encode the Image Line Bit Pattern?
Yes
Bit Assignment Function = External
No
Bit Assignment Function = Default
122
Bit Mask Indicator

FIG. 20

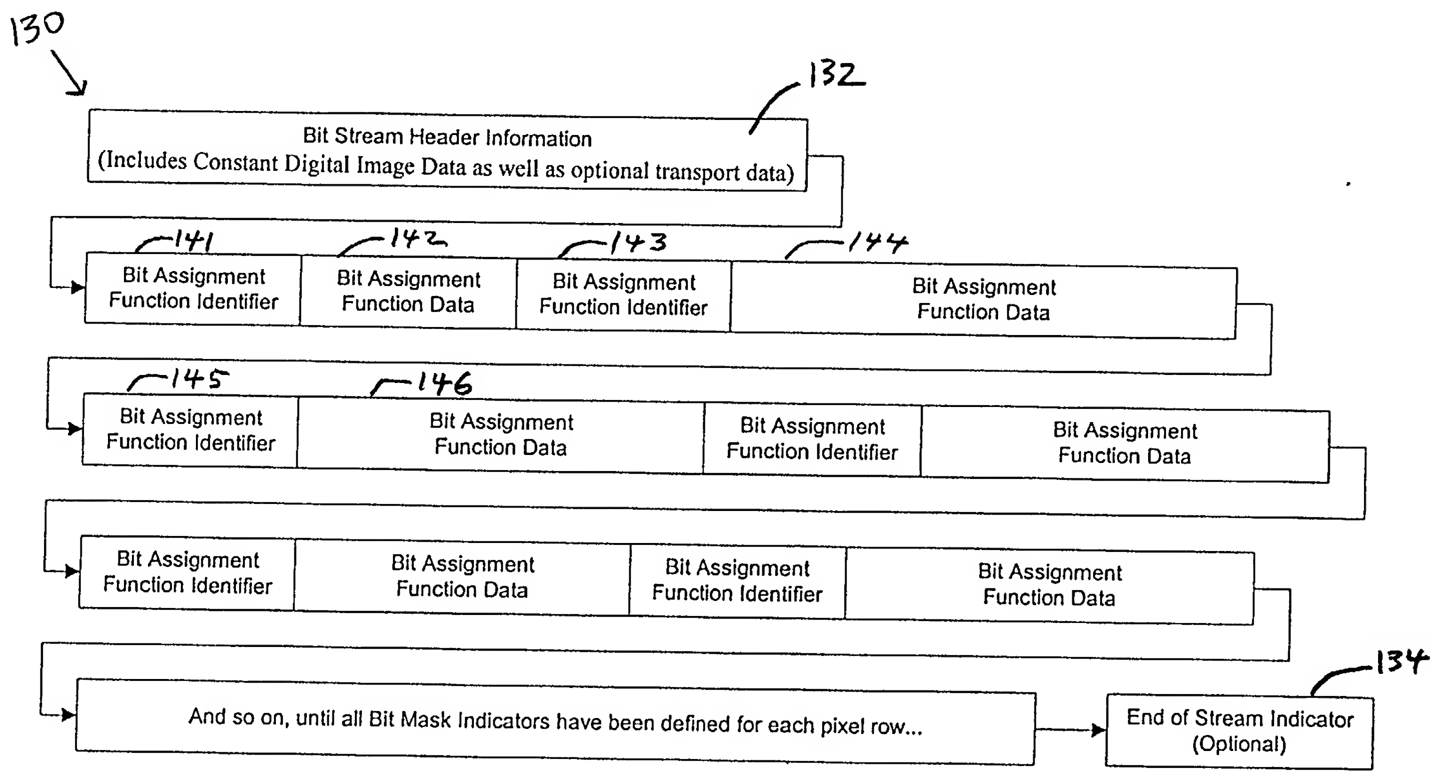
130
132
Bit Stream Header Information
(Includes Constant Digital Image Data as well as optional transport data)
141
142
143
144
Bit Assignment Function Identifier
Bit Assignment Function Data
Bit Assignment Function Identifier
Bit Assignment Function Data
145
146
Bit Assignment Function Identifier
Bit Assignment Function Data
Bit Assignment Function Identifier
Bit Assignment Function Data
Bit Assignment Function Identifier
Bit Assignment Function Data
Bit Assignment Function Identifier
Bit Assignment Function Data
And so on, until all Bit Mask Indicators have been defined for each pixel row...
134
End of Stream Indicator
(Optional)

FIG. 21

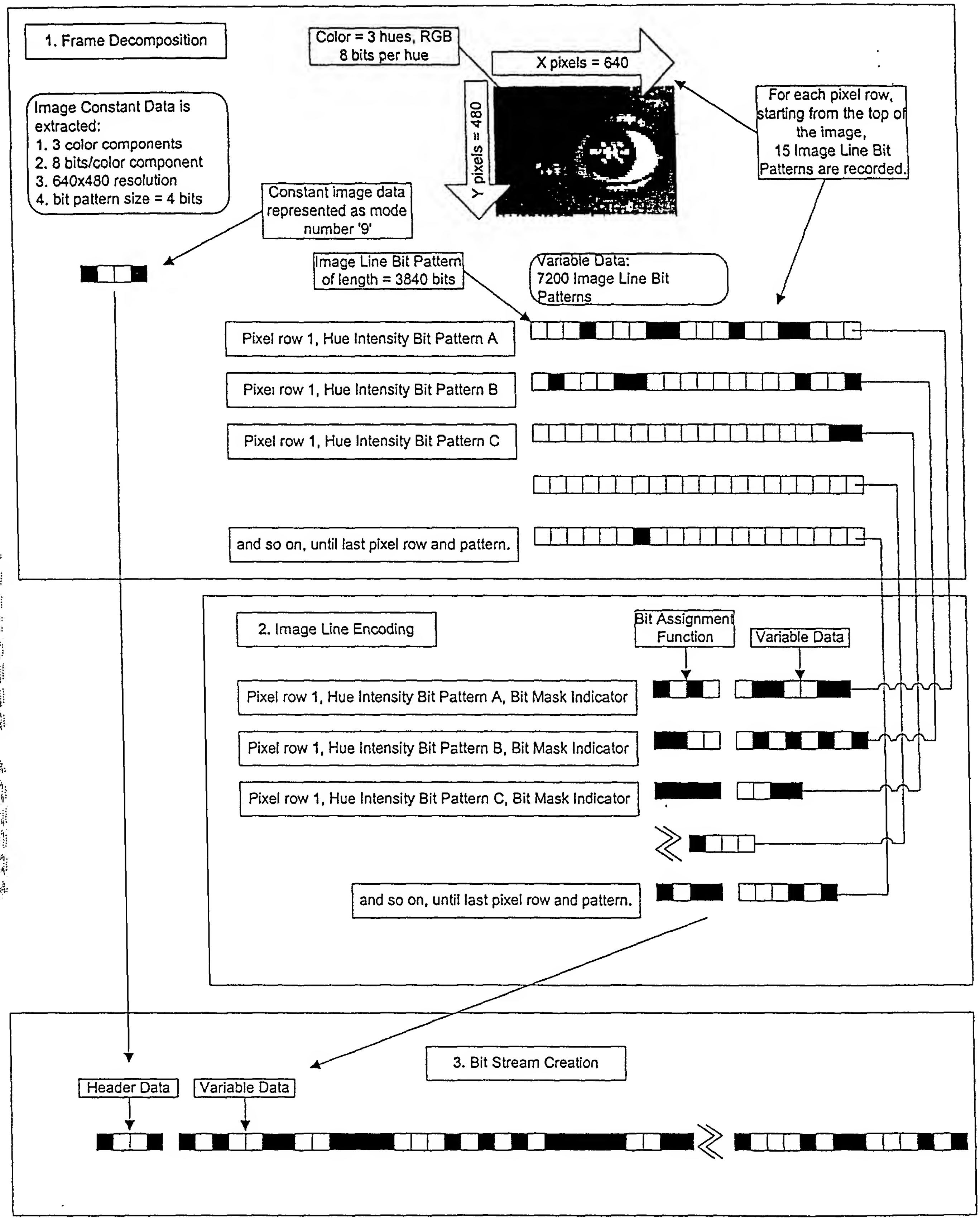
1. Frame Decomposition
Color = 3 hues, RGB
8 bits per hue
X pixels = 640
Y pixels = 480
For each pixel row, starting from the top of the image, 15 Image Line Bit Patterns are recorded.
Image Constant Data is extracted:
1. 3 color components
2. 8 bits/color component
3. 640x480 resolution
4. bit pattern size = 4 bits
Constant image data represented as mode number '9'
Image Line Bit Pattern of length = 3840 bits
Variable Data: 7200 Image Line Bit Patterns
Pixel row 1, Hue Intensity Bit Pattern A
Pixel row 1, Hue Intensity Bit Pattern B
Pixel row 1, Hue Intensity Bit Pattern C
and so on, until last pixel row and pattern.
2. Image Line Encoding
Bit Assignment Function
Variable Data
Pixel row 1, Hue Intensity Bit Pattern A, Bit Mask Indicator
Pixel row 1, Hue Intensity Bit Pattern B, Bit Mask Indicator
Pixel row 1, Hue Intensity Bit Pattern C, Bit Mask Indicator
and so on, until last pixel row and pattern.
3. Bit Stream Creation
Header Data
Variable Data

FIG. 22

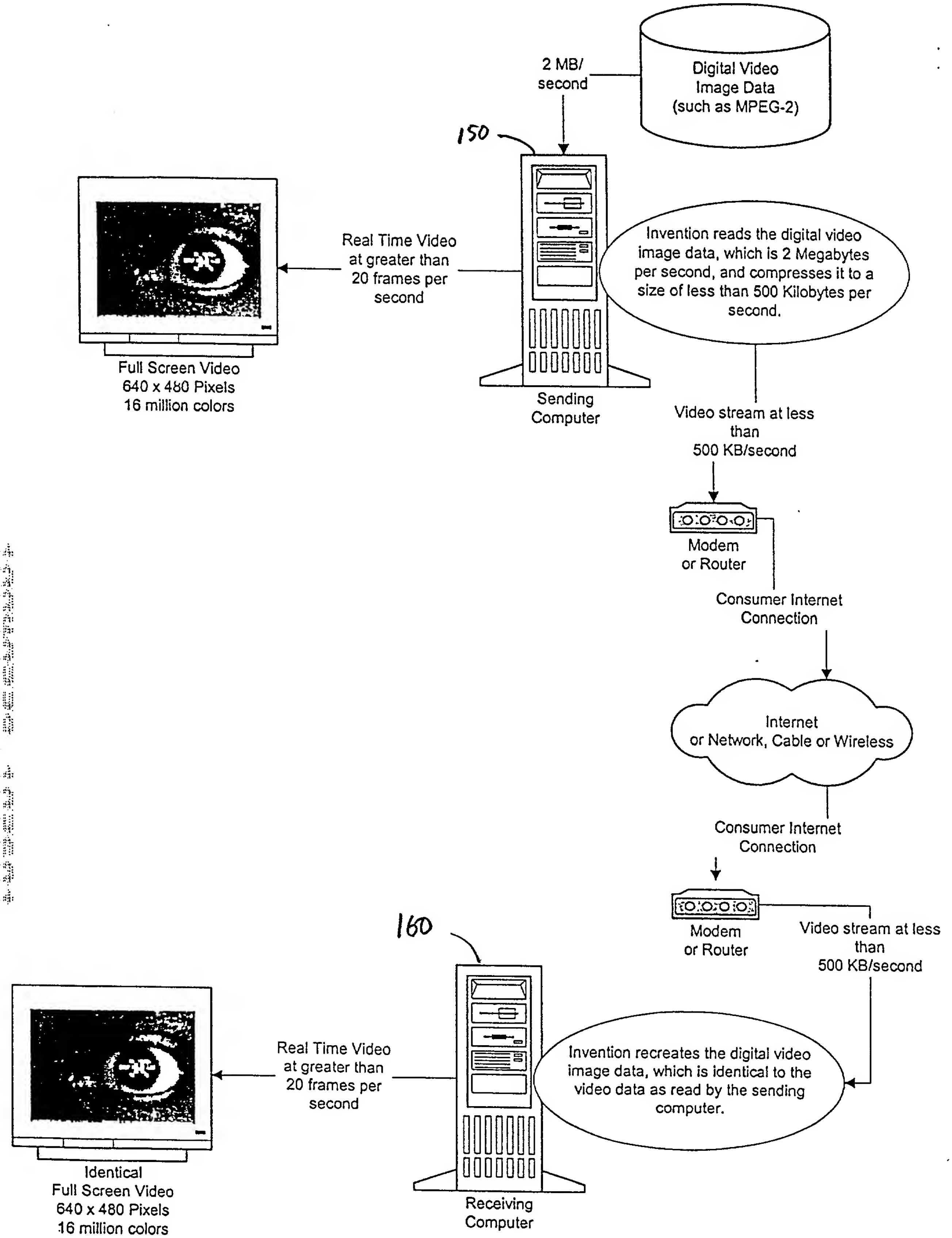
2 MB/
second
Digital Video
Image Data
(such as MPEG-2)
150
Real Time Video
at greater than
20 frames per
second
Invention reads the digital video image data, which is 2 Megabytes per second, and compresses it to a size of less than 500 Kilobytes per second.
Full Screen Video
640 x 480 Pixels
16 million colors
Sending
Computer
Video stream at less
than
500 KB/second
Modem
or Router
Consumer Internet
Connection
Internet
or Network, Cable or Wireless
Consumer Internet
Connection
Modem
or Router
Video stream at less
than
500 KB/second
160
Real Time Video
at greater than
20 frames per
second
Invention recreates the digital video image data, which is identical to the video data as read by the sending computer.
Identical
Full Screen Video
640 x 480 Pixels
16 million colors
Receiving
Computer

FIG. 23

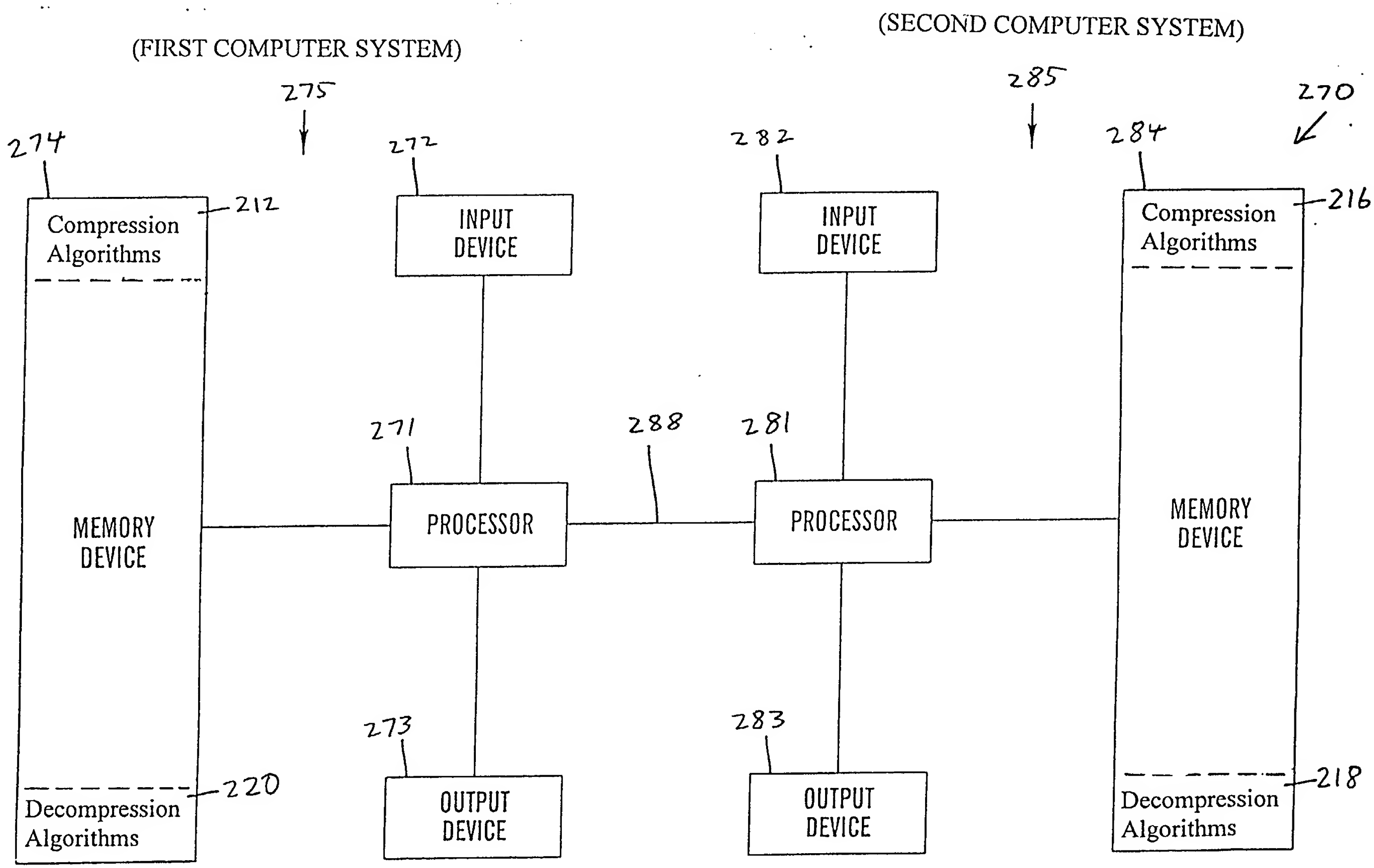
(FIRST COMPUTER SYSTEM)
(SECOND COMPUTER SYSTEM)
275
285
270
274
272
282
284
212
216
Compression
Algorithms
INPUT
DEVICE
INPUT
DEVICE
Compression
Algorithms
271
288
281
MEMORY
DEVICE
PROCESSOR
PROCESSOR
MEMORY
DEVICE
273
283
220
218
Decompression
Algorithms
OUTPUT
DEVICE
OUTPUT
DEVICE
Decompression
Algorithms

FIG. 24